MHEG 5 Frequently Asked Questions

This FAQ is based on the former MHEG-5 Verification Groups FAQ and the MUG FAQ from 1997. It was revised end of 1997. If you have something to add (questions or answers) please send a mail.

■ What is MHEG ?

MHEG is an acronym for Multimedia and Hypermedia Experts Group. This group develops within ISO several standards, which deal with the coded representation of Multimedia and Hypermedia Information.

■ What is the correct meaning of the acronym "MHEG"?

MHEG is an acronym for "Multimedia and Hypermedia Expert Group." However, the acronym is also used to refer to a standard with the name "Coding of multimedia and hypermedia information." This may be counter-intuitive, but also similar to MPEG

■ What is MHEG-5?

MHEG-5 is the fifth part of the MHEG standard suite. It was developed to support the distribution of interactive multimedia applications in a client/server architecture across platforms of different types and brands. The standard defines a final-form representation for application interchange so that the application only has to be developed one time.

■ Is there a simple introduction to MHEG-5?

Yes. It is a bit outdated (DIS Version), but still good for a quick introduction with some substance. See the documents section for this.

■ Can I get the spec of MHEG-5 (i.e the IS document) ?

Due to regulations of ISO finished documents like the MHEG-5 IS (International Standard) are protected by copyright. Therefore, there is no public text available. You have to contact ISO or your national ISO authority (e.g. ANSI in the US, DIN in Germany, AFNOR in France etc). Ask for ISO 13522-5.

■ How do I pronounce MHEG ?

The German pronunciation of MHEG is indeed em-haeck, and the French one does sound like "meg". The English one, however, I think is "Emm Hegg". And don't forget the Dutch one: "Emm Hech" and the Swedish one: "Emm Hegg" (sing it!).

■ How can I make an MHEG-5 engine? Are there any tools or publicly available source codes?

Design of an MHEG Engine is subject of current work. At this time it is quite unlikely to find sourcecode of MHEG engines in the public domain. Check out the links in the links section.

■ Is it possible to get some MHEG-5 objects in order to test my MHEG-5 engine?

If there are contributions they will be published at the MUG site. In the mean time check the links in the links section.

■ Are there any authoring tools available for MHEG-5 objects?

There is a number of tools available. See the links in the links section.

■ I have a script. Can I use my script in the MHEG-5 engine?

You have to ensure, that the MHEG engine supports the script language and the required facilities for execution. MHEG-5 does not imply to use a certain script engine or language. However, the use of a Java VM together with the Java bytecode is encouraged by MHEG-6.

■ Where can I download a draft of MHEG 5 or 6 document?

Please contact your ISO representative.

■ Why does a STU need the MHEG-5 engine? What's the advantage to have the MHEG-5 engine inside a STU?

An MHEG-5 engine on an STU insures that MHEG-5 applications will run on that STU. This allows the development of applications only one time and have that application run on multiple platforms. The advantage of having a runtime engine as part of the STU is that:

- The runtime does not have to be downloaded.
- The engine could be optimized (hardware support, etc.)

The disadvantage is that:

- The engine is harder to upgrade
- The standard is still maturing

■ What kind of games can I play using the MHEG-5 engine?

Several types of games have been built on a MHEG-5 like runtime engine. This include card and board games, mystery games, and trivial answer games. But there is not much experience with building fast twitch games like you find in arcades.

■ What type of applications is MHEG-5 for?

MHEG-5 was designed for interactive multimedia applications. This included but is not limited to: Video on Demand, Home Shopping, Games, Education, and Informational. The standard allows large applications to be distributed between server and client This allows the client to have a much smaller memory need.

■ How can I join the MHEG users group/mailing list?

There is a majordomo-based mail replicator installed. Detail are at the site credits page. You may contact the maintainer of this site if every other thing fails.

■ Why doesn't the standard define content types?

It does. But the type checking is quite weak, which moves certain responsibility to the user of MHEG.

■ What network bandwidth is needed for a typically MHEG-5 application?

That depends heavy on the type of content data and not on the use of MHEG.

■ Why do you need MHEG-5 when there is Java?

The Java language, the Java virtual machine and the Java runtime environment combined form the set of specifications that make it possible to write applets that can be embedded in HTML documents "Web pages". The target environment for MHEG-5, however, is an interactive TV terminal, which is quite different from the typical environment of a Web browser. This difference manifests itself in two ways in the MHEG-5 specification: the focus on declarative rather than procedural code, and the effort to keep the footprint size of the engine to a minimum. The focus on declarative code provides for easier off-line "static" testing and validation of application behaviour. The requirement for a small footprint is an obvious result of the extremely constrained memory size of interactive TV terminals

■ Why do you need MHEG-5 when there is HTML?

HTML and MHEG-5 have many concepts in common, such as the focus on declarative code. However, HTML is inherently a document description language, not a format for describing multimedia applications. MHEG-5 is built from the ground up with the needs of multimedia applications in mind, such as synchronization and speed control of streams, handling of stream events etc. Therefore, MHEG-5 is ideally suited to express applications in the interactive TV domain, which HTML is not. Secondly, HTML is a very dynamic standard, which is ideal in the Internet environment. However, for a consumer electronics product such as an interactive TV terminal, stable interface definitions and guaranteed compliance are of prime importance. MHEG-5 aims to provide that.

■ What is the footprint size of a typical MHEG engine?

It is difficult to give exact figures on this head. However, an MHEG-5 engine which can rely on reasonable graphics, user interface management, and communications support from its platform can certainly fit in a few hundred kB.

■ What is ASN.1? What is DER?

ASN.1 (Abstract Syntax Notation One) is a widely-used ISO standard that allows the description of a data structure's syntax in a standardized, platform-independent format. Once a syntax has been described in ASN.1, public-domain tools can be used to automatically generate code that translates internal data structures into an exchangeable binary format, and back. ISO defines several version of that exchangeable format MHEG-5 uses one of them called DER, Distinguished Encoding Rules. The DER define bit-by-bit the final form exchange format used for MHEG-5 applications (alternative: Textual notation).

■ Is anybody actually building MHEG-5 engines?

A list of active projects will be available in the "links" page. The entries there are definitively not complete or the full picture. Some companies have also banners on the banners page.

■ How can MHEG applications be made?

MHEG-5 defines a final-form exchange format for "bits on the wire" and therefore does not say anything about the language or environment used to author the applications. Possible approaches to authoring MHEG applications are either to type it in in a textual pseudo-code provided by MHEG, or to develop a backend to an existing authoring environment. There are some commercial products available.

■ Is there a home page for MHEG where I can get more information about other parts of the standard and general information about the MHEG standards work?

Yes. Try http://www.demon.co.uk/tcasey/wg12.html
